



Performance Analysis in Support of Capability Computing (CC) on Red Storm/XT3

Mahesh Rajan
Sandia National Laboratories

First Annual Cray Technical Workshop--North America February 26-28,
2007 Gaylord's Opryland Hotel Nashville, Tennessee

Sandia is a multi-program laboratory operated by Sandia Corporation, a Lockheed Martin Company,
for the United States Department of Energy's National Nuclear Security Administration
under contract DE-AC04-94AL85000

3/1/2007 2:07:35 PM

Cray Workshop; Nashville;
2/26/07-2/28/07

Motivation


- DOE push with CC campaign
- Sandia Director level initiative to pave the way for analyst to effectively use CC
- Red Storm GA LVL II milestone scaling studies
- Compare alternate HPC systems for CC and understand challenges on the path to Petaflops computing

Presentation Outline

- Background information on CC with Red Storm and Purple
- CC examples, their impact and lessons learnt
- Platform comparisons and lessons for Petaflops class simulations
- Use of Performance Analysis Tools

Background: DOE CC campaign objectives

- To ensure that capability system resources are allocated on a priority-driven basis according to the program requirements
- To utilize ASC capability systems for large capability jobs for which they were designed and procured
 - Similar to large experimental facility like telescopes
 - Large jobs with supporting smaller jobs – several smaller calculations often building up in size are executed in support of one or more large calculations
- Job categories
 - A single job 75% or more of the nodes – may be dedicated time
 - 30-75% of the nodes; may be 2 cc runs simultaneously
 - < 30% -- preparatory work towards larger CC



Background: Capability Computing Challenges

- Complex tasks with multiple stages and must cross a few hurdles to ensure success
- Relatively new experience for most analysts and multi-disciplinary team often needed
- Performance issue mostly focused on scaling of the application to thousands of processors but other stages could become a time sink
 - Model prep
 - Model slice and spread
 - I/O at all stages including execution
 - Large file transfers for visualization
 - Parallel visualization with large number of files (1/PE)

Background: Red Storm and Purple efforts

- Red Storm – May, June 2006 – before upgrade
- Applications
 - SIERRA/Fuego – WIAF
 - SIERRA/Presto – B61 – Head-on
 - ITS – Neutron generator Design
 - CTH – Vulnerability analysis
 - Alegria/HEDP – Z-pinch
- Purple – started
 - code teams working on a similar set of applications
 - Support team supporting and investigating scaling of CTH, Salinas, Fuego, Presto

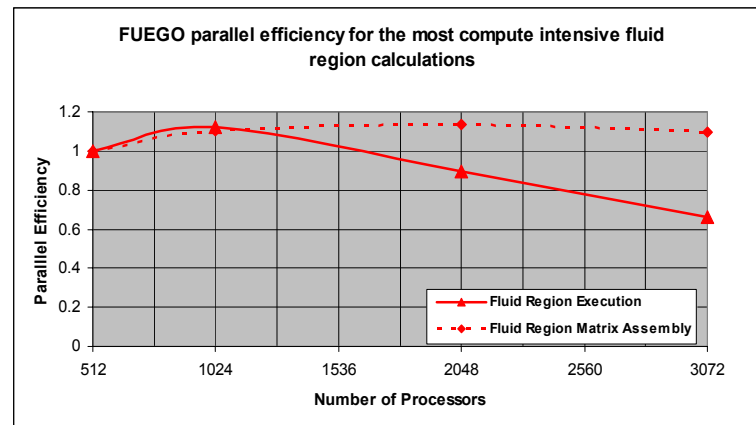
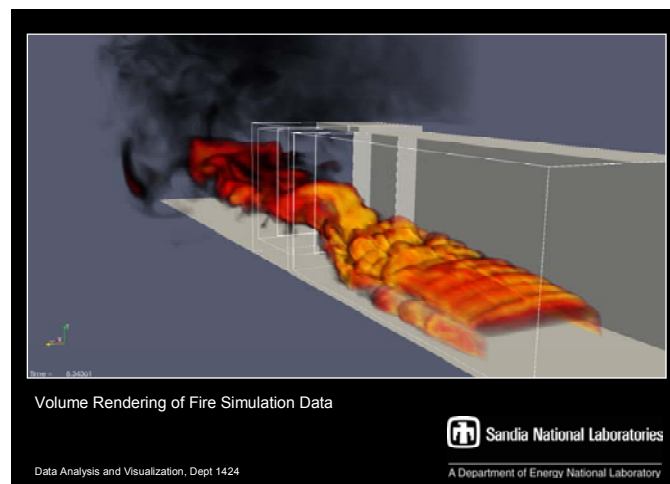
Examples: Red Storm Capability Simulation Accomplishments

- **W76-1 Safety:** Red Storm simulations requiring 5000 processors for one week were a key element in fire testing for the qualification of the W76-1
- **B61-7 Crushup:** Using 5000 nodes of Red Storm, unexpected results with potentially significant consequences have been uncovered in crush-up and buckling simulations of the B61-7
- **LANL MZONE-classified:** LANL received 56 days of Red Storm time using 5000 nodes, with a computer-use efficiency of 89% during the second quarter 06
- **W78 Neutron Generator:** In support of qualification efforts for the W78 Neutron Generator (Quantitative Margins and Uncertainty for Thermo-Mechanical Shock), ITS radiation transport calculations

Red Storm Capability Example -1

SIERRA/Fuego

- WIAF (abnormal thermal env.) simulations
 - 385 Million degrees of freedom
 - Multi-physics; Fluid reacting flow, participating media radiation, enclosure and calorimeter conduction.
- Red Storm simulation is the first with sufficient resolution to allow Large Eddy Simulation (LES) techniques — rather than Reynolds Average Navier Stokes (RANS) “engineering” methods
- Long-runs (days) on 5000 nodes to achieve quasi-steady state
- Simulation results used for accreditation of the code for safety analysis and guide experimental tests



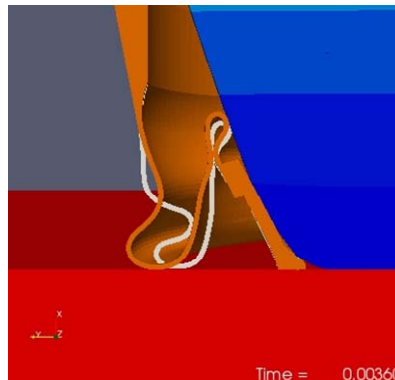
Red Storm Capability Example -2

SIERRA/Presto

- Presto B61-7 Head-On Impact
 - Design analysis for assessing weapon surety.
 - In the past, computational resources imposed meshing limitations that mandated the use of sub-scale models and engineering adjustments
 - 32 M element model
 - Contact algorithm computationally demanding; runs several days with restart on 5000 nodes

B61 PRESTO Results for 3.6 msec into Impact on 5000 nodes of Red Storm

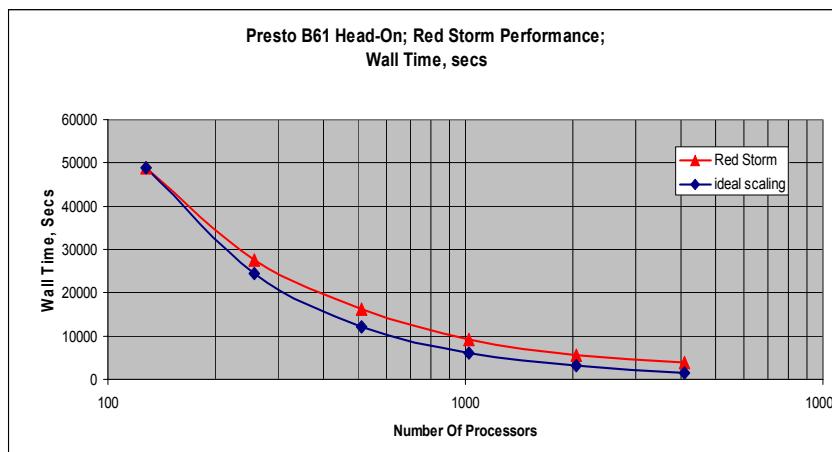
Buckle formation earlier on finer model (white) due to greater flexibility of finer mesh



Load will be limited up the outer wall after the buckle forms, further limiting load transfer to case in wider portion of nose.

Centercase components will be required to take more load during impact with finer model.

Changing load path of the structure is a fundamental change between the coarse and fine models – exactly the purpose of this exercise.



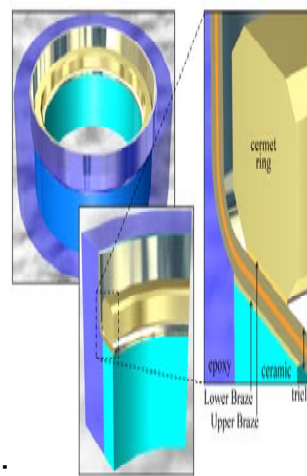
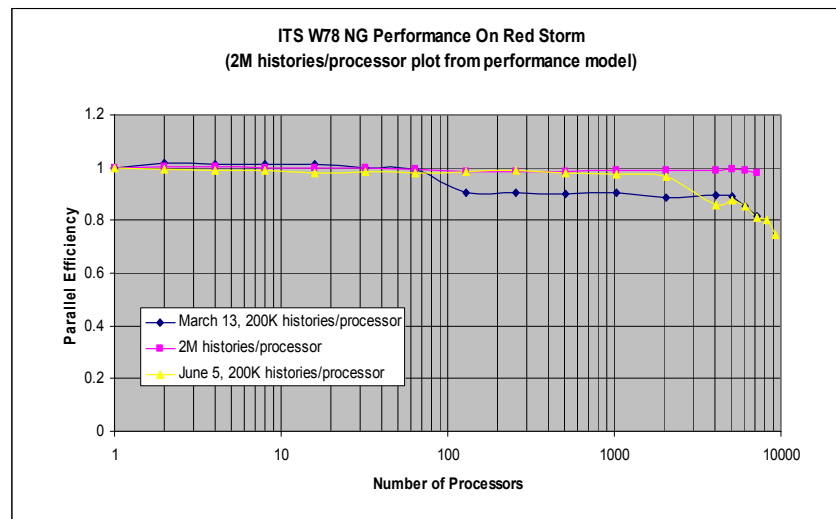
3/1/2007 2:07:35 PM

Cray Workshop, Nashville;
2/26/07-2/28/07

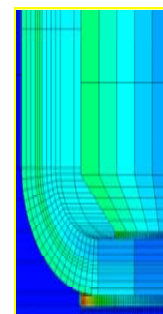
Red Storm Capability Example -3

ITS

- ITS, Monte Carlo radiation transport code
- Dose distribution computed and passed to Presto, where thermo-mechanical shock is calculated
- In support of qualification efforts for the W78 Neutron Generator
- 120 billion X-ray “histories”
- CAD model with 4 million “sub-zones”
- energy resolution more than double previous maximum were computed
- 9 hour run time on 7500 processors at 96% parallel efficiency to obtain a 3-D dose spatial profile
- For the first time, we were able to simulate energy deposition over the full geometry of the neutron generator, enabling realistic structural analyses that have the potential to uncover problems.



3D Dose Spatial Profile



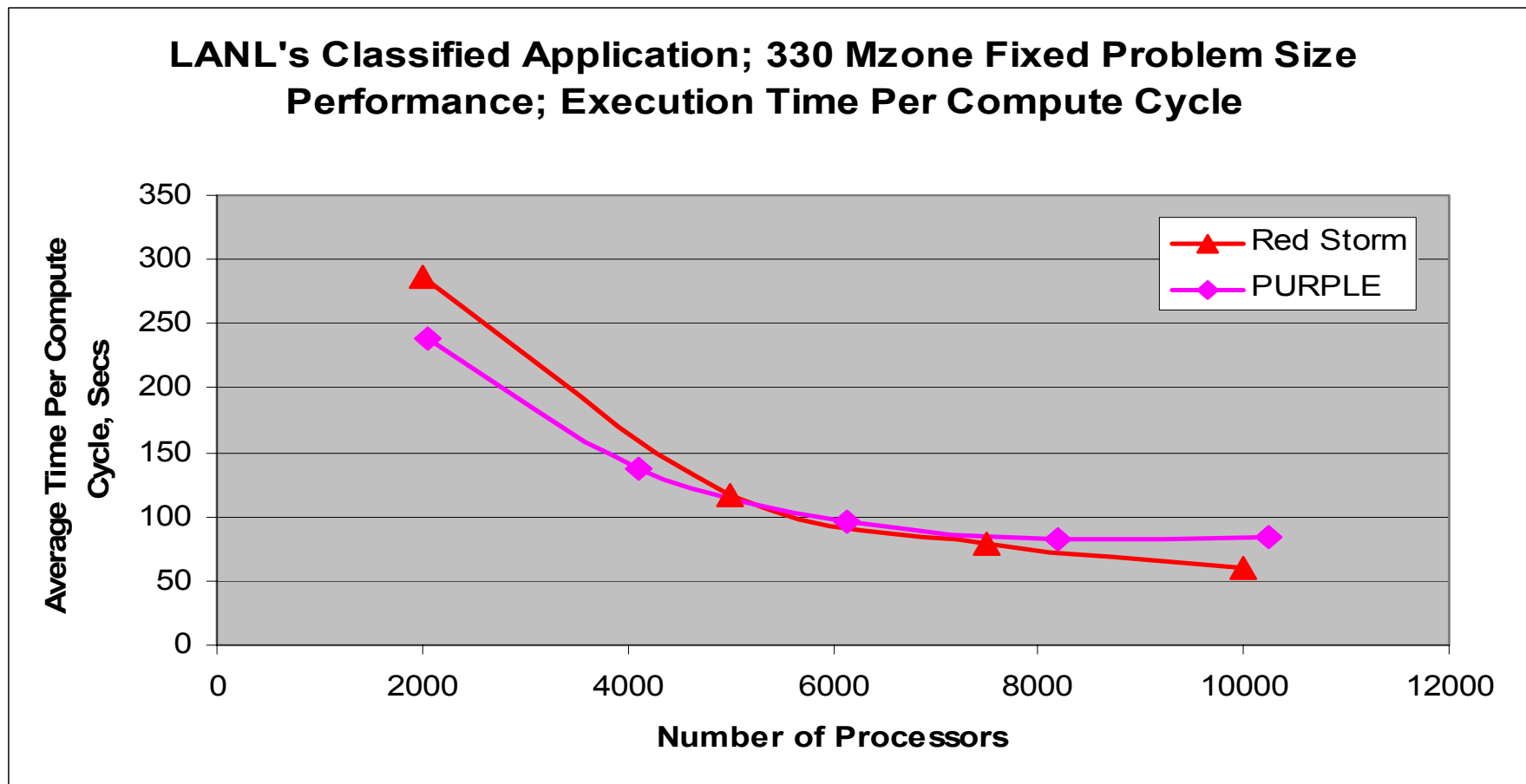
Lessons Learned From capability runs and scaling studies

- Fuego – Scaling study supported mesh convergence study
 - ML Solver scalability limit identified and ML solver team engaged
 - File open/restart issues identified
 - Data movement of 5000+ large files to Rose
 - SIERRA Framework vs ML Solver scaling compared
- Presto – Scaling study supported B61 Lay-down DSW milestone
 - Limitations of tools needed for large mesh generation
 - Several releases compared to help development team identify optimal for capability runs
 - Paraview visualization with large number of files
 - Sierra job launch script problems with queues and with scripts
 - gmalloc needed for job termination and performance improvement
 - I/O – optimal disk stripe count and iobuf improvements
- ITS – Scaling study supported W78 Neutron Generator qualification investigations
 - CAD model read&spread performance improved after initial scaling study

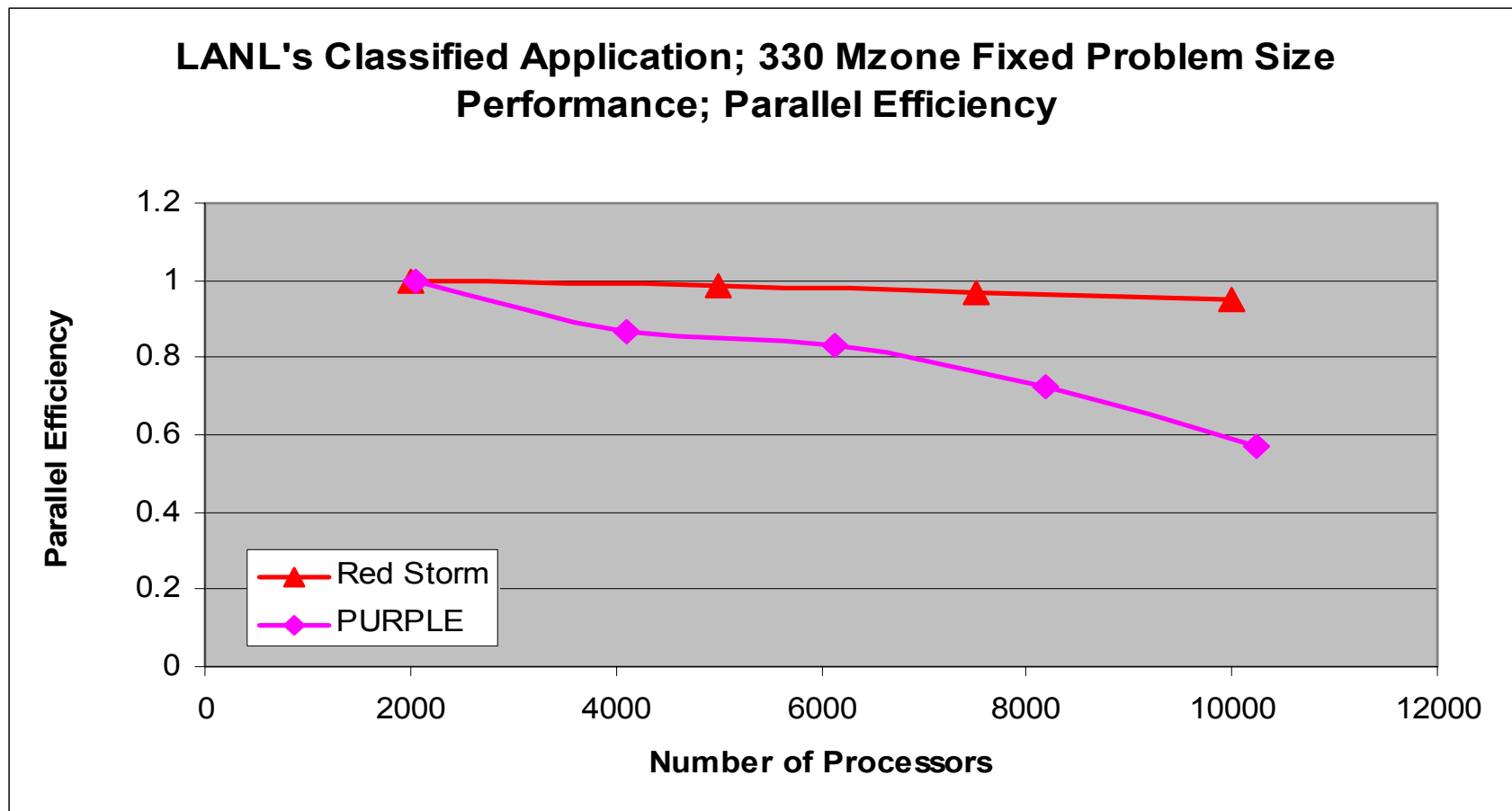
Red Storm, Purple comparison

Item	Red Storm, AMD-Opteron (before upgrade)	Purple, IBM-P5
Clock Speed	2.0GHz	1.9GHz
L1; L2; L3	64 KB(I,D); 1 MB; None	32KB(D)-64KB(I); 1.9MB; 36MB per chip
Memory	2GB (3GB, 4GB)	32 GB/node of 8 PEs
SPECint	1370	1398
SPECfp	1389	2850
LINPACK HPC/ PE	3377 GFLOPS	7084 GFLOPS
Stream Triad	4707	7198
Inter node MPI Latency, BW	6.5us, 1.15 GB/s	4.4us, 3.4GB/s

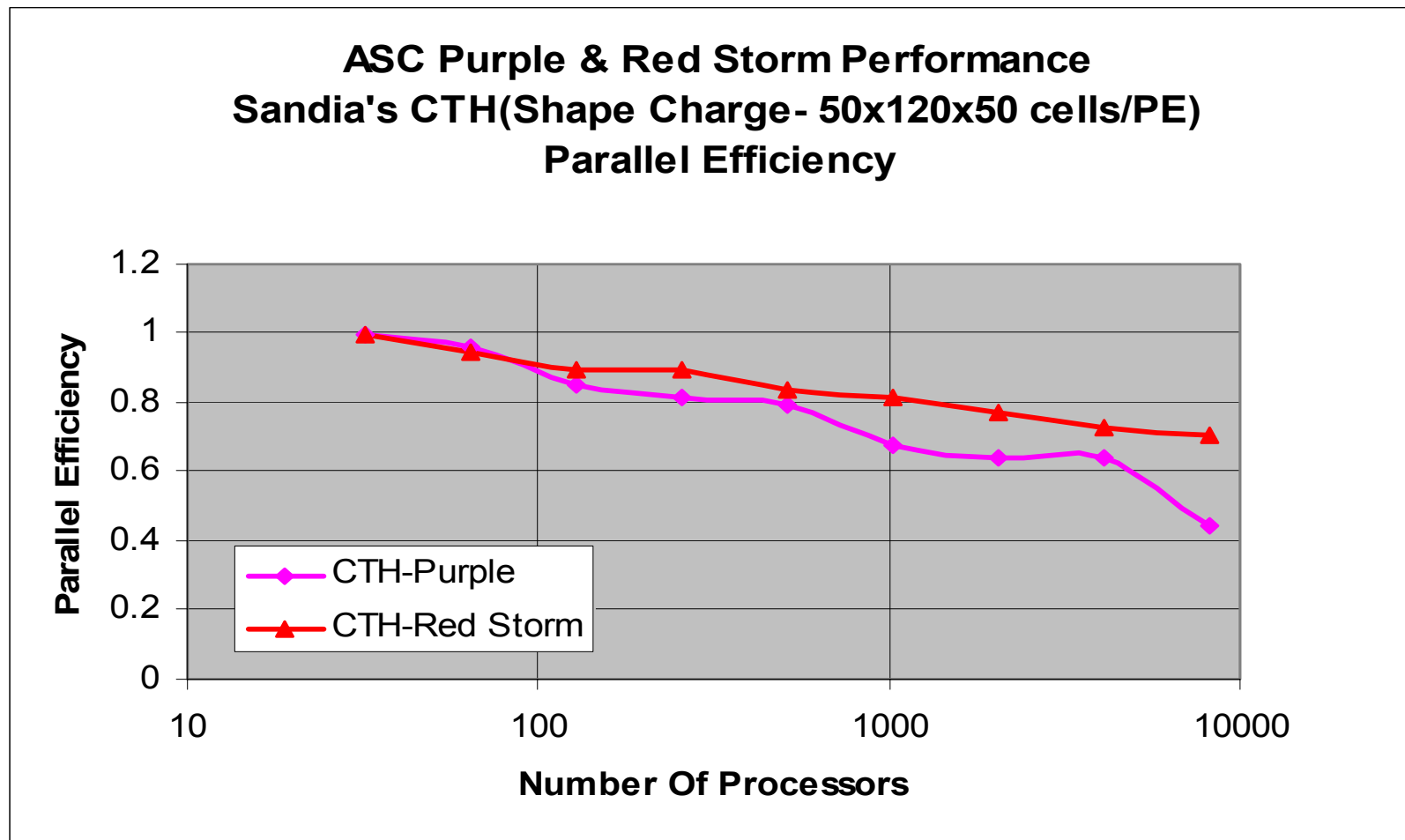
Performance Comparisons: LANL's classified 3-D "A-to-B" physics parameter comparison model; (from John Daly)



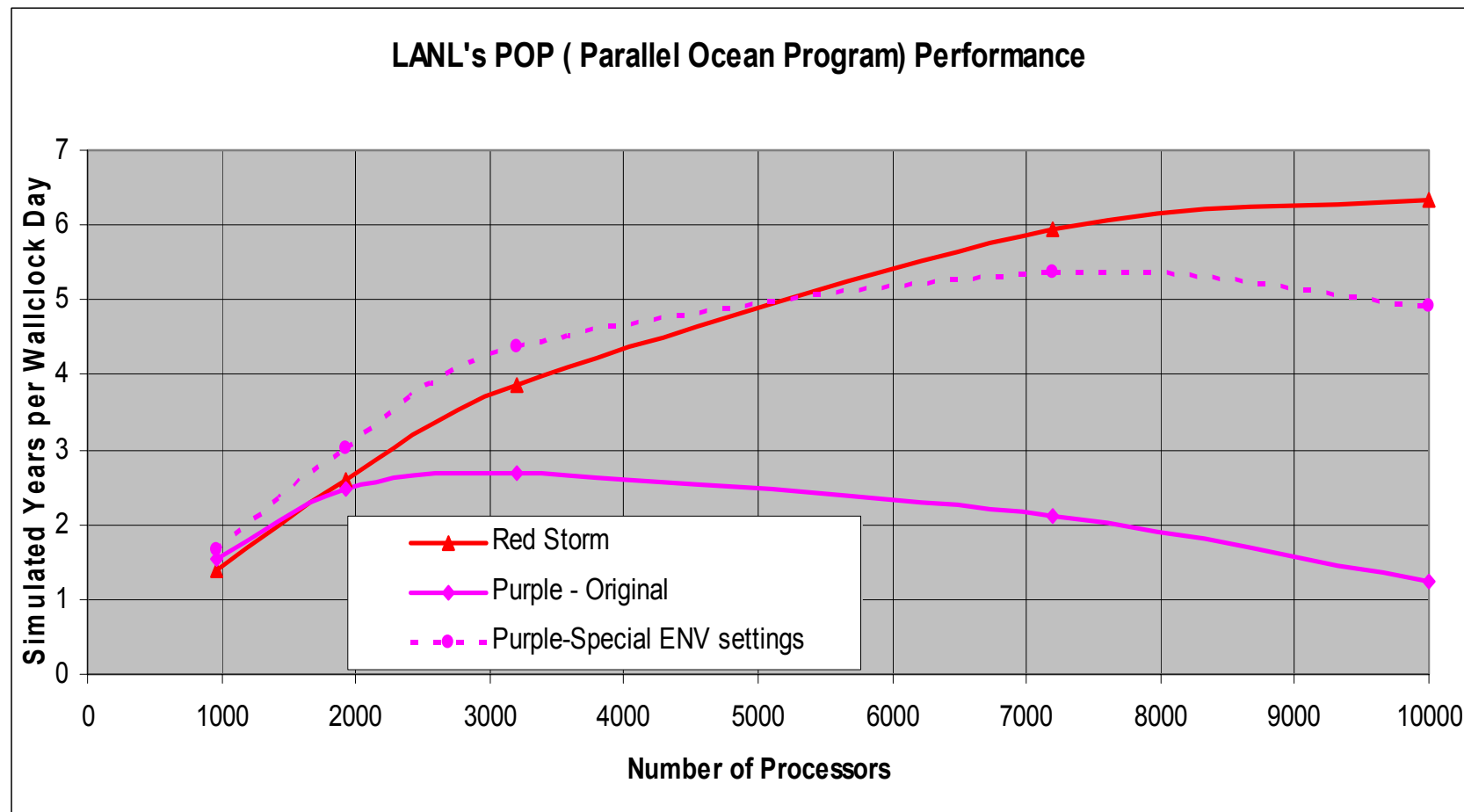
Performance Comparisons: Red Storm Shows better scaling than Purple on LANL's classified 3-D "A-to-B" physics parameter comparison model (from John Daly)



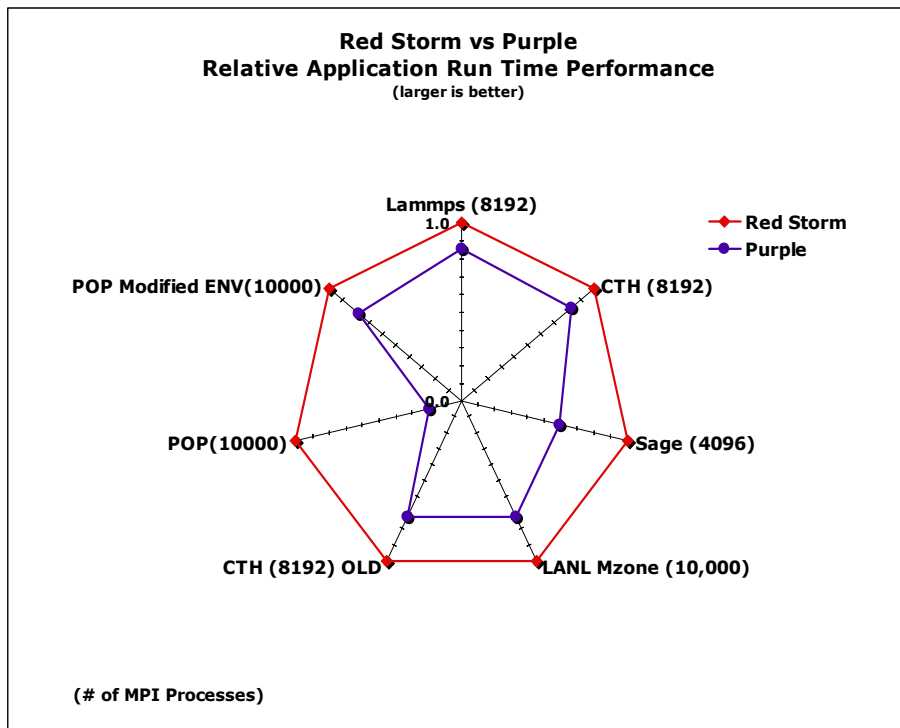
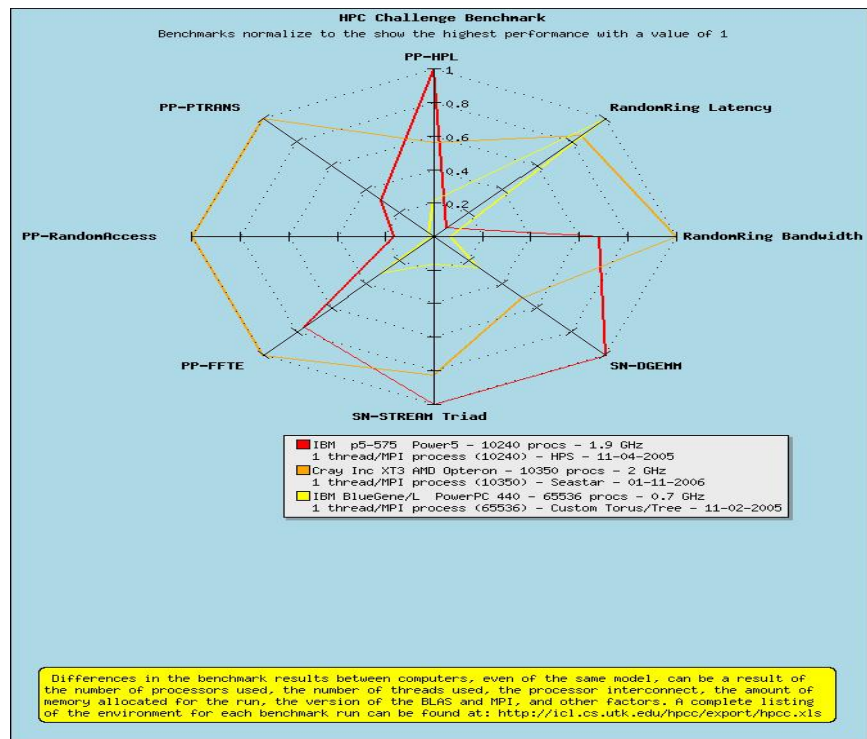
Performance Comparisons: CTH Sandia Hydrodynamics Code for Shape Charge Input



Performance Comparisons: LANL's Ocean Modeling Code POP; 1 block/PE, CG solver, global grid=3600x2400



Large Capability Simulation: platform comparison



- Application performance on a system is determined by many factors, chief among which are processor architecture, the memory subsystem architecture, and the interconnect architecture.
- The operating system may also have a big impact on scalability. Applications differ considerably on how they may stress each of the above.
- The HPC Challenge benchmark suite compares performance with measures for floating point compute power (HPL Linpack), memory subsystem performance (STREAMS), and interconnect performance (GUPS, FFT, PTRANS, RandomRing).
- Red Storm's superior architectural balance seems to be the prime differentiator.



Performance Comparison: Lessons Learnt

- For large node counts run times may vary by 50 % on Purple due to OS noise
- For some applications like POP with lots of global operations between time steps the MP_POLLING_INTERVAL interrupts can destroy application scaling
- OS Noise impact seems to have a significant impact on other applications at 4k, 8k node counts
- 4 flops/clock did not result in 2X performance gain seen when compared with 2 flops/clock
- Large memory SMP node is very useful for many pre/post processing operations

Conclusions

- SIERRA/Fuego – solver scaling issue identified leading to LDRD by the solver team – additional inefficiency in mesh transfer between different regions noted
- ITS – cad geometry distribution improved – earlier modeling effort lead to better parallel efficiency though algorithmic modification for statistics computation
- POP – performance inhibitor on Purple traced to MPI Polling – work around with env setting
- Noise effects seen in large simulations (CTH, SAGE) on Purple
- Exodus large model for seacas tools to work properly (64/32 bit tool issue)
- Nem-slice takes too long for large meshes even on large SMPs
- Movement of data for large files resulting from capability simulation can be sometimes painful and time consuming
 - Data corruption may not be spotted till one attempts to use it
 - May take days to complete
- Remote viz problems (vizto purple)
- Other seacas Post processing utils like nem-join (needed for viz when one has restart to complete long runs)
- Pre-processing utilities like cubit when used for large mesh preparation like automatic creating of 8X models may be very slow since their basic operation is serial

Recommendations

- Thread-parallelize wherever possible in pre/post processing tools
- Understand sweet spot for each application and each problem set for optimal use of CC cycles
- Large CC runs may reveal inhibitors like
 - Noise effect on collective operations
 - I/O inefficiency (like lfs_setstripe use on RS)

Use of Performance Analysis Tools

- Evaluate use of CrayPat
 - Sparse Matrix solver
 - ICARUS DSMC Code
 - LAMMPS Molecular Dynamics Code
- Try to quantify the gap between peak performance and sustained; It is widening
 - Multi-core architecture racing ahead of concurrency
 - Memory bottlenecks

Use of Cray_pat to understand performance; Mike Heroux's' Sparse Matrix CG solver to compare GFLOPS

Num. Of PEs	64	128	256	512	1024	2048
Code Inst. GFLOPS	15.3	28.0	56.3	111.8	224	472
PAPI Measure: GFLOPS	13.4	24.8	49.8	98.8	197	450

PAT_RT_HWPC=1 used in tests

Comparison gives confidence in use of Cray pat for GFLOPS count

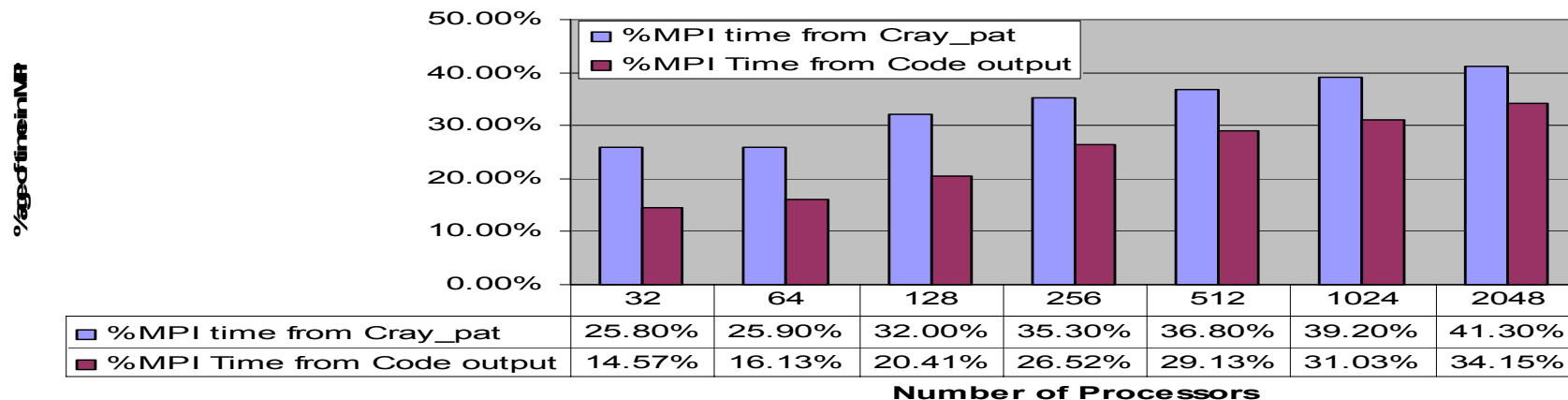
pat_report version 3.0 fails at > 2048 PEs

Release 3.0 creates directory and one .xf file per PE, 3.1 combines into one file, but not as robust as 3.0
yod -VN used (both core used in test)

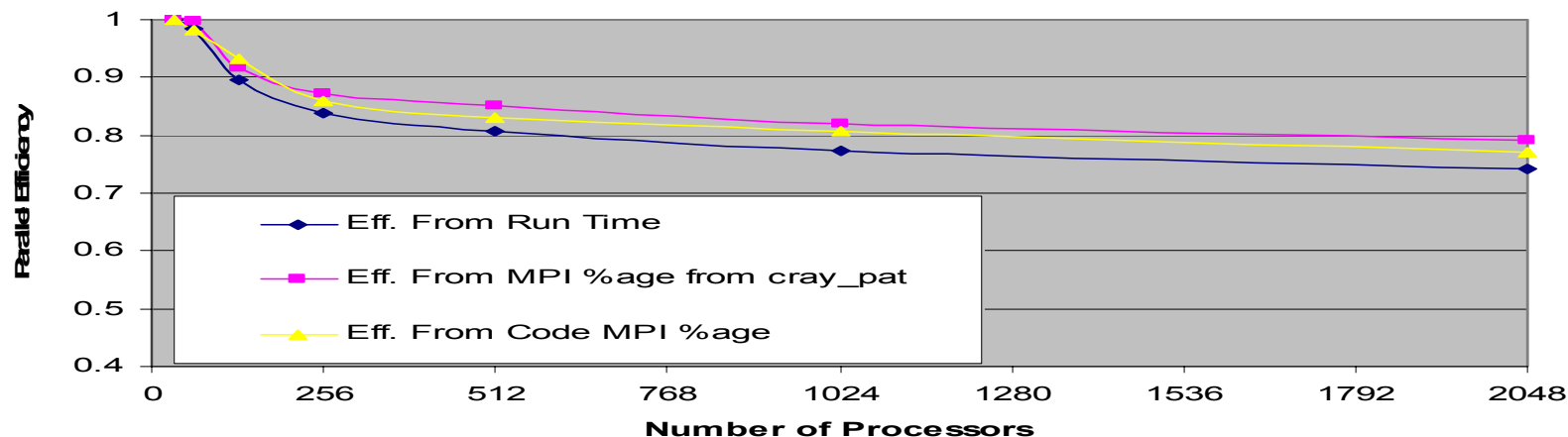
~11% difference; cray_pat measure includes other setup times not accounted in code instrumentation

“pat_build -g mpi” useful for modeling/analysis ICARUS-DSMC MEMS (low density flow) Code

ICARUS-DSMC Communication time Percentage (1 ppn)

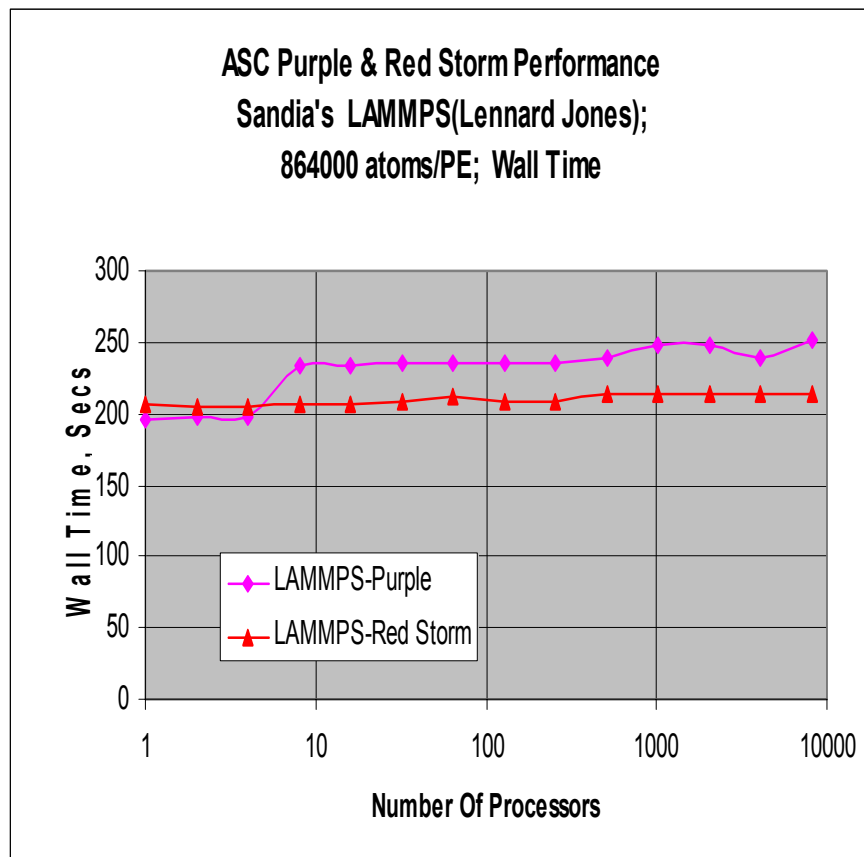


Parallel Efficiency of ICARUS-DSMC MEMS code (1 ppn)



Use of CrayPat to understand parallel scaling;

LAMMPS; molecular dynamics code



Num. PEs	%MPI time (CrayPat)
32	1.5
64	2.1
128	1.5
256	2.1
512	1.8
1024	2
2048	2.4

Use of CrayPat/PAPI to understand performance

LAMMPS; ICARUS; SAGE

	LAMMPS – 1PE	ICARUS – 32 PE	SAGE – 1PE
Total Cycles	438334149027	23022223454	146844868231
Total Instructions	322843999218	855901630778	70445132839
Floating point ins.	159193963401	154415317565	12306664167
Branch Instructions	18035055818	76266692411	6753460052
Run time	182.63 secs	9.592 secs	61.18 secs
MFLOPS	871 (18.2%of peak)	16097 (10.5%of peak)	201.13 (4.2% of peak)
%Floating point Ins.	49%	18.04%	17.47%
%branch Ins.	5.6%	8.9%	9.58%
Computational intensity	0.92 ops/ref	0.51 ops/ref	0.42 ops/ref

Acknowledgement

This work would not have been possible without the collaboration of the following individuals. Thanks!

Sue Kelly, Courtenay Vaughan, Hal Meyer, Mark Taylor, Amalia Black, Stefan Domino, Sheldon Tieszen, Terry Hinnerichs, Tim Jones, Joel Stevenson, Mike Davis, Brian Franke, Tom Laub, Nathan Crane, John Daly (LANL), Tom Spelce (LLNL), Jeff Fier (IBM), Ken Gwinn, Garth Reese, Riley Wilson, Luba Kmetyk